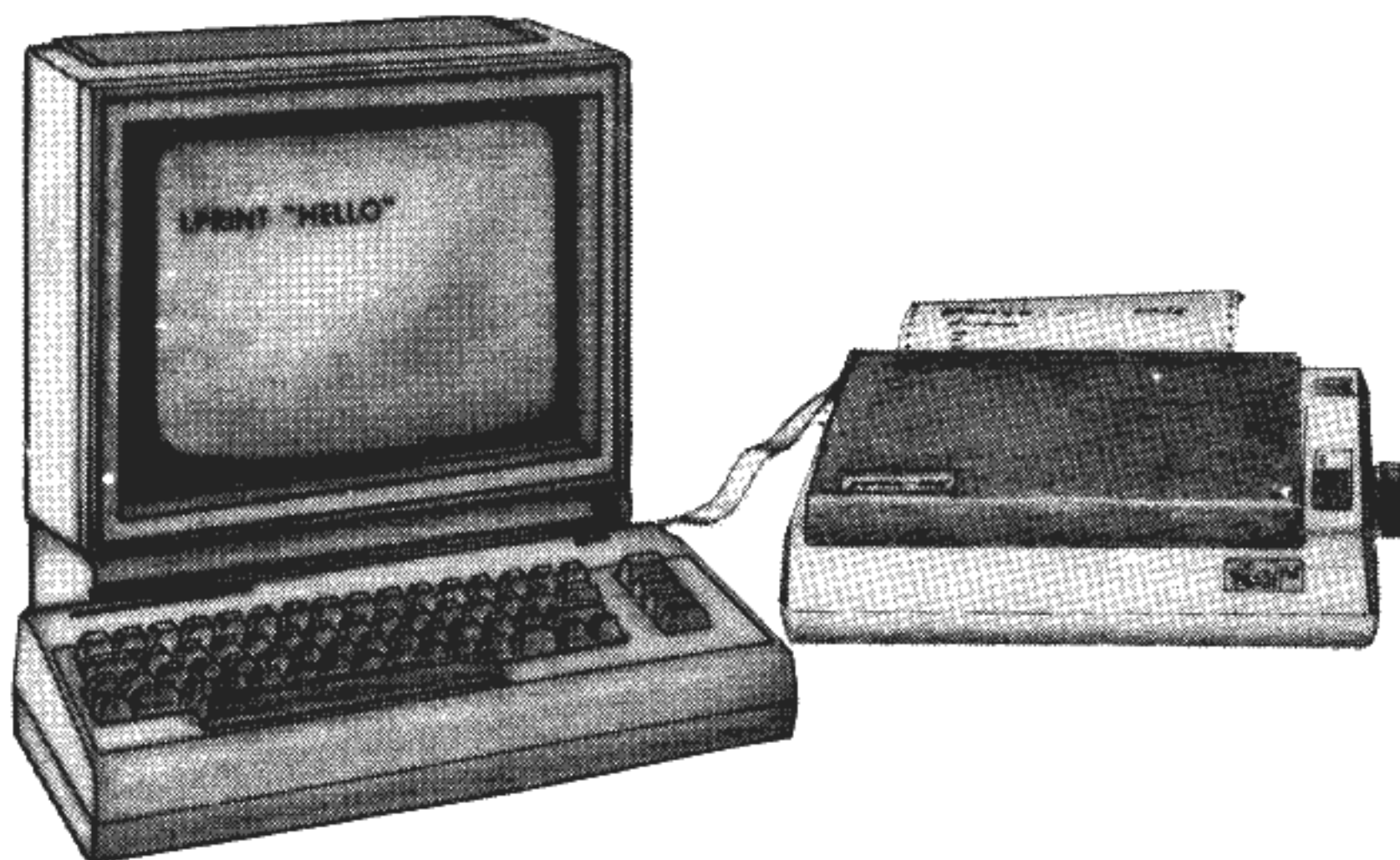




GRAPHICS PRINTER INTERFACE

FOR THE VIC 20 AND C-64



INSTRUCTION MANUAL

NOTES:

FIRST EDITION

Copyright (C) 1984 by Xetec Inc.
All rights reserved.

All specifications subject to change without
notice.

TABLE OF CONTENTS

Introduction	1
Setting it up.	1
Configuring your interface	2
Switch settings for different printers	3
Making it print.	3
OPEN and CLOSE	6
PRINT#	7
CMD.	7
Secondary addresses.	8
Command channel.	9
Auto line feed	12
Transparent mode	13
Hexadecimal output	14
Normal printing.	14
Listing programs	15
Locking the interface.	16
Print width.	16
Pagination	17
Other special features	17
TAB.	18

APPENDICES

A - Troubleshooting.	19
B - Special graphics codes	20
C - Listing abbreviations.	22
D - Parallel cable pinout.	23
E - Interfacing non-standard printers.	23
F - Changing the device number	24
G - Programs	24

SAMPLE PROGRAMS

PRINT# demo (character set).	7
CMD example.	8

Line feed test12
Tab example.18
Miscellaneous.24
Screen dumps25

INTRODUCTION

Xetec's Graphics Printer Interface (GPI) is designed to allow the Commodore VIC 20 or C-64 to print with any parallel printer that is Centronics compatible. It is primarily intended as a substitute for the Commodore 1525 printer, and therefore can be configured to make your printer work just the same. It has special functions, however, which the 1525 cannot perform. If you plan to make use of these features, a thorough reading of this manual is suggested. Otherwise, a skim will probably get you up and running.

If you run into any problems, call us at (913) 827-0685 Monday through Friday, 7:00 to 5:30 CST.

SETTING IT UP

- 1) Turn off the computer and printer.
- 2) Plug the round six-pin connector into the serial port at the back of the computer. (If you are using a disk drive, plug it into the unused port on the back of the drive.)
- 3) Plug the large connector at the end of the flat cable into the parallel input port on your printer.

- 4) Plug the connector on the end of the single wire into the cassette port with the white connector on the bottom and the silver side of the circuit board on top (viewed from the back, the wire should be closer to the left side). To use the cassette drive, plug it onto the board you just plugged in.
- 5) Turn on the printer first, then the computer.
- 6) Make it print:

```
Type: OPEN 4,4  
Then: PRINT#4,"IT WORKS!"  
      CLOSE 4
```

If it does, you have successfully setup your interface. If not, refer to appendix A, troubleshooting.

CONFIGURING YOUR INTERFACE

The six switches visible through the hole in the upper-right corner of the interface determine how your interface should operate. Switches 1, 2, and 3 control printing options and will be discussed later. Switches A, B, and C select the type of printer you have attached to the interface. Once you set these three switches, you should not need to change them again (unless, of course, you change printers). Find your printer in the following list and set the switches accordingly.

PRINTER

SWITCHES

	A	B	C
Banana	on	on	off
C-Itoh 8510	off	on	off
Delta 10	off	off	off
Epson	off	off	off
Gemini 10-X	off	off	off
Gemini 15	off	off	off
GX-100	on	on	off
NEC 8023	off	on	off
Okidata 82	on	off	off
Okidata 92	on	off	off
Prowriter	off	on	off
Siekosha 100	on	off	off

If your printer is not in the list above, read appendix E, 'NON-STANDARD PRINTERS.'

NOTE: The switch settings are only checked when the interface is powered-up. Therefore, any changes you make to the switches will not affect the interface until it is reset. To do this, either turn the computer off and on, or unplug the connector with the single wire and plug it back in.

MAKING IT PRINT

If you intend to use only purchased software to drive the printer, you can safely skip this section. If, however, you want to use the printer from your own programs and take advantage of some special features of your interface, read on.

Before you can write programs that use your printer, you need to familiarize yourself

with a few BASIC commands which may be new to you. If they are, learning them may be a painful process. To help ease the pain, imagine this scenario: at your house you have 127 phones (all different lines, of course!). If you want to talk to someone, there are a few steps you must take. First, you must decide which of your many phones you will use. Next, you must dial the number of the person you want to talk to. Finally, you talk all you want, after which you hang up the phone. So what has this got to do with your printer?!? Well, in many ways, your computer talks to your printer the same way the person above calls people on the phone. You will see the similarity if you type the following lines:

```
OPEN 20,4
PRINT#20,"HELLO"
PRINT#20,"THERE"
CLOSE 20
```

The first line tells the computer to use 'phone' number 20 (out of the 127) to 'call' whoever might have a 'phone number' of 4. Your interface comes from the factory set up to respond to 'phone number' 4. What that means is that in the first line above, you 'called' the printer, and it is now waiting for you to 'talk' to it. So in the next two lines, the words 'HELLO THERE' are sent through phone 20 (the one you called the printer on). The last line tells the computer that you have no more information to send, so it should hang up phone 20 (you don't even have to say goodbye!).

Every peripheral that you hook up to the

serial bus (printers, disk drives, etc.) must be given a different 'phone number' (actually, they are called device numbers, not phone numbers). Printers are usually assigned to device numbers 4, 5, 6, and 7, with 4 being the most common. Therefore, if you attach two printers to your computer, you might give one device number 4 and the other 5. Then when you 'call' a printer with the OPEN command, if you specify device number 4 it will be printed on one printer, and 5 will print on the other. Appendix F details how to change the device number from 4 to about anything you want.

Since you have 127 'phones' (actually called 'files'), you obviously can be talking on more than one of them at once. In fact, you can even call the same device on two different 'phones' at the same time. You are limited, however, to ten 'conversations' at one time. As an example, the following program will do the same thing as the one above, but through two different files (phones):

```
OPEN 3,4
OPEN 2,4
PRINT#2,"HELLO"
PRINT#3,"THERE"
CLOSE 3
CLOSE 2
```

Now that you understand the mechanics of OPENing a file to the printer, a more technical discussion of the BASIC commands is needed.

OPEN and CLOSE

The format for the OPEN statement is:

OPEN <file>,<device>

or

OPEN <file>,<device>,<secondary address>

The 'file' number can be any from 1 to 127, although only ten can be open at once. You are permitted to use file numbers 128 to 255, but keep in mind that they stick an extra line feed in for each carriage return, so using these file numbers will most likely cause your printouts to be double-spaced.

The device number must be the one which was selected by the interface switch (normally '4'). If you have more than one printer attached, this number will determine to which printer the data should be directed.

The optional 'secondary address' is used to send commands to your printer interface which tell it how to handle the data that will be coming to it through the file. If 'secondary address' is not included, a '0' will be sent. See page 9 for a listing of all the secondary addresses and their functions.

Once a certain file is opened, it cannot be opened again until closed. Notice that if you alter a BASIC program, all open files are closed. Once a file has been opened, you may talk all you want to the printer until you CLOSE it.

The format for the CLOSE statement is:

CLOSE <file>

where 'file' is the number of the file to be closed.

PRINT#

Once a file has been opened to the printer, data may be sent to it the same way you would print data to the screen - with the PRINT statement. The only exception is that when you PRINT to the printer, you must tell the computer which file to talk through (which phone to talk on). The format is therefore:

PRINT#<file> ,

To get an idea of how to use this statement, try this program:

```
10 OPEN 1,4
20 PRINT#1,"CHARACTER SET"
30 PRINT#1
40 FOR I=32 TO 90
50 PRINT#1,I;"=";CHR$(I)
60 NEXT I
70 CLOSE1
```

A few things about the format: there must be no space between 'PRINT' and the '#', and you cannot abbreviate it with '?'. In other words, '?#1,' will not work for 'PRINT#1,'.

CMD

There is another way to send data to the

printer. Once a file has been properly opened to a printer, using

CMD<file>

will direct all subsequent output to the printer instead of the screen. In other words, a normal PRINT statement will print to the printer, as will a LIST. To return the output to the screen, use

PRINT#<file>

If you forget to do this and just try to close the file, usually all output will still go to the printer. Try this example to become familiar with 'CMD.'

```
10 PRINT "CMD EXAMPLE"
20 OPEN 1,4
30 PRINT#1,"THIS IS PRINTED"
40 CMD 1:REM DIVERT TO PRINTER
50 PRINT "THIS IS PRINTED TOO"
60 PRINT#1:REM DIRECT BACK TO SCREEN
70 CLOSE 1
```

SECONDARY ADDRESSES

The following is a list of the valid secondary addresses that may be used in the OPEN command (the x in OPEN 4,4,x) along with their functions.

COMMAND	FUNCTION
0	Normal printing, upper case only with line feed
1	Normal printing, upper case only without line feed
3	Hexadecimal output
4	Transparent, with line feed
5	Transparent, without line feed
7	Normal printing, upper/lower case with line feed
8	Normal printing, upper/lower case without line feed
20+ any above	Locks the interface in any of the above modes.
15	Command channel

COMMAND CHANNEL

If you open a file with secondary address 15 (eg. OPEN 1,4,15) you have gained access to the command channel. When you send data through this channel, it gets intercepted. Instead of being sent to the printer, it is interpreted as a command. The legal commands are as follows (page numbers refer to the command's explanation in the text):

COMMAND	FUNCTION
---------	----------

G	*List graphics characters as graphics (p. 16)
A	List graphics characters as their ASCII values (p. 16)
K	List graphics characters as their keystrokes (p. 16)
M	*List control codes as mnemonics (p.15)
V	List control codes as inverse graphics (p. 15)
U	Unlock interface (undo a 20+ command) (p. 16)
6	*Space text at 6 lines-per-inch
8	Space text at 8 lines-per-inch
R	Reset interface (and read switches)
E	Emulate VIC 1525 printer (p. 17) [Short for "V6GP"CHR\$(0)"W"CHR\$(80)]
N	Interface to non-standard printer (p. 23)
Dx	Set new device number (p. 24)
Px	Set pagination (default 0) (p. 17)
Wx	Set width (default 0) (p. 16)

LN Turn off auto line-feeds (p. 12)
LY Turn on auto line-feeds (p. 12)
LS *Line-feeds software-selectable (p. 12)
TY Make interface transparent (pp. 13,14)
TN *Shut off transparent mode (p. 14)
IY Print all characters from internal
 character generator (p. 17)
IN *Use printer's character generator for
 text (p. 17)
C Credits

 * Default modes on power-up

 As an example, if you want to emulate the
VIC printer, and want to print 8 lines-per-
inch, you might use

```
OPEN 1,4,15  
PRINT#1,"E8"  
CLOSE 4
```

 Any number of the commands may be sent in
any order with no syntax to follow (the "E8"
above could have been "E 8", "8E", "8 E", or
even "OOZE! 78" - illegal characters are
ignored).

 If you OPEN two files to the interface,
one with secondary address 0 and the other
with 15 (eg. OPEN 1,4,0:OPEN 2,4,15), then
when you want to print data, use PRINT#1,...,
to send a command, use PRINT#2,...

AUTO LINE FEED

As you can see from the table on page 9, the secondary address you use determines whether the interface should send an auto line feed or not. Which you use will most likely depend on your printer. When a carriage return code is sent at the end of a line, some printers just move the head to the left margin, while others also move the paper up a line (do a line feed). To check your printer, try this program:

```
10 OPEN 4,4,1
20 PRINT#4,"IT DOES      SEND A LINE FEED"
30 PRINT#4,CHR$(13)CHR$(13)CHR$(13)CHR$(13)
40 PRINT#4,""TAB(7)"N'T"
50 PRINT#4,CHR$(10)CHR$(10)
60 CLOSE 4
```

If it told you that it doesn't send one, then you are in great shape. You are free to choose secondary addresses with line-feeds for normal printing, or without them to doublestrike a line, superimpose letters, etc. You may override a secondary addresses' line-feed mode with the command channel. For example, if you have a file presently OPENed with secondary address 7, you can shut off the auto line-feed without closing and re-opening in 8 by sending 'LN' through the command channel. Sending 'LY' will force auto line-feeds to be enabled regardless of the secondary address. 'LS' returns to the normal mode, making line-feeds software selectable (determined by the secondary address).

If the program above indicated that your

printer does send an auto line feed, check your printer manual to see if you can shut it off (on most you can). If so, do it and all the options stated in the preceding paragraph will apply. If you can't shut the line feed off (did you remember to turn the printer off and on?) then you will want to disable line feeds in your interface (you don't want both the printer and interface sending one, or you will get everything doublespaced). To do this, flip switch 2 to the off position. With this switch off, there is no way to enable the auto-line feed function (not even with the command channel). You can, however, still send line-feeds manually [using CHR\$(10)]. Remember that you will have to reset your interface (either by powering-up again or by sending "R" through the command channel) to recognize the change in the switches.

Secondary address 3 (Hex. output) sends auto line feeds and has no equivalent command that doesn't send one. If you followed the above steps correctly, it should work on your printer just the way it is.

TRANSPARENT MODES

Secondary addresses 4 and 5 (or 'TY' in the command channel) put the interface in the 'transparent' mode, which means that as codes are received from the computer, they are relayed to the printer with no modification. The interface may be locked 'transparent' by shutting off switch 3 (ASCII correction).

When sending special commands to your printer (to change print width, underline text etc.) you must be careful because some of

these commands have special meanings to your interface and therefore will probably not work properly. To avoid this conflict, send commands to the printer in the transparent mode. If you don't want to re-open a file with secondary address 4 or 5, you may use 'TY' (to the command channel) to temporarily make the interface act transparent. Use 'TN' to change it back.

HEXADECIMAL OUTPUT

Secondary address 3 puts the interface in the hexadecimal output mode. Used for debugging your print routines, this mode will print every code it receives as its hexadecimal value. For example, if you OPEN file 1 to the printer (with secondary address 3) and type PRINT#1,"HI" this mode will print -48-49-0D which are the hexadecimal ASCII values for 'H', 'I', and carriage return. When a carriage return is received, a new line of ASCII values is started.

NORMAL PRINTING

Normal printing modes are the ones that you will probably use most often. They are 0, 1, 7, and 8. The only differences between the modes are whether to send an auto line-feed and the case (upper/lower or upper only).

In the upper-case-only modes, the codes sent from the computer are interpreted as text, control characters, or graphic characters. In the upper/lower case modes, however, some of the graphic characters are now interpreted as the lower case alphabet.

In other words, if you want to print in upper and lower cases, you will have to sacrifice some of the graphics characters normally available.



In the normal printing modes, the interface makes some modification to the incoming data to make your printer act like the VIC printer. See appendix B for a complete description of all the recognized special graphics commands and code conversions.

Sending CHR\$(17) (or the cursor down key) will put the interface into the upper/lower case mode. CHR\$(145) (or the cursor up key) puts it into the upper-case-only mode. Both of these codes are compatible with the VIC printer.

LISTING PROGRAMS

To LIST a program to the printer all you need to do is open a file, use 'CMD <file>' to divert output to the printer, and type LIST. Graphics characters in a BASIC listing can be printed in several ways. First of all, control codes such as 'CURSOR LEFT' and 'HOME' can be printed as an inverse graphic character (the way it appears on the screen) or can be listed as a mnemonic ([LEFT] and [HOME]). Since the latter is more readable, it is the default mode on power up. See appendix C for a listing of all the mnemonics supported. Send 'V' to the command channel to select the inverse graphic method. 'M' puts you back in the mnemonics mode.

Graphics characters can be listed in one of three ways. The default method is to print the actual graphic character. Alternately,

they may be printed as their commodore ASCII values (eg [213][207]). The final method is to print the keystrokes that produced the character. For example,  is represented as [SW] meaning SHIFT-W, and  is printed as [C*] meaning C= * (the commodore key and '*'). Commands G, A, or K to the command channel (see page 10) selects which of the three methods to use.

LOCKING THE INTERFACE

By using secondary addresses 20 through 28, you can lock the interface in modes 0 through 8. For example, to lock in the upper/lower case mode (7), use number 27 (20+7). Once locked in this manner, the only way to unlock it is to power down or by using 'U' in the command channel. It may be necessary to lock the interface in a certain mode before starting a word processor.

PRINT WIDTH

It is possible to limit the width that lines are to be printed. If, for example, you want to emulate the VIC printer on a wide printer, you'll need to specify a width of 80 characters or the printing may not come out as it would on the 1525. To specify a width (from 1 to 255) send this command to the command channel

```
PRINT#<file>,"W"CHR$(width)
```

If you specify a width of 0, the interface

will not care about the width anymore (this is the default mode).

PAGINATION

Your interface also has the ability to break its output into pages. To turn this mode on, send `PRINT#<file>,"P"CHR$(pagelength)` to the command channel. This tells the interface how long your pages are (number of lines). It will print four blank lines at the bottom of each page to skip over the perforation. To shut this mode off again, specify a page length of 0. Also, when you give it a page length, it thinks it's starting at the top of a page, so you should position the printer that way. As you can see, the 'P' is also used to set the top of form (just resend the same page length).

OTHER SPECIAL FEATURES

The 'E' command to the command channel is a combination of five other commands that cause the interface to exactly emulate the VIC-1525: `"V6GP"CHR$(0)"W"CHR$(80)`

The command 'IY' causes even text to be printed from the interface's internal character generator. This has the drawback of being a little slower than if the printer handles it. If, however, graphics characters and text characters are not coming out the same width on your printer, you may need to invoke this mode so that they will be. To shut the mode back off, send 'IN'.

TAB

The TAB function works a little different with a printer than you are used to. First of all, PRINT#1,TAB(5) will give a ?SYNTAX ERROR. This seems to be a bug in the Commodore computers. The only way it will work is

```
PRINT#1,"TAB(5)
```

Notice the similar statement in the sample program on page 10.

If your printer automatically sends its own line feeds, you are out of luck. The only way for you to simulate tabs is to print a computed number of spaces to go from the current to the next column.

If your printer doesn't send its own line feeds, you can use the TAB function by opening the file without the line feed option. The only difference is that to advance to the next line, you must send a CHR\$(10).

```
10 REM TAB EXAMPLE
20 OPEN 1,4,1
30 PRINT#1,"TAB(5);5TH"
40 PRINT#1,"TAB(20);20TH"
50 PRINT#1,"TAB(10);10TH";CHR$(10)
60 CLOSE 1
```

APPENDIX A Troubleshooting

<u>SYMPTOM</u>	<u>POSSIBLE CAUSES</u>
Computer locks up when trying to print	<ul style="list-style-type: none">- Printer not connected- Printer deselected- Printer off
Doesn't lock up but nothing prints	<ul style="list-style-type: none">- Wrong device#, switch- Printer off or desel.
Device not present error	<ul style="list-style-type: none">- Serial cable disconn.- Wrong device#, switch
File not open error	<ul style="list-style-type: none">- Wrong file number- BASIC program altered
Line of data writes over previous line	<ul style="list-style-type: none">- Using command 1,5,8- Switch 2 (LF) off
Spaces between each line	<ul style="list-style-type: none">- Using command 0,3,4, or 7 with switch 2 on, and a fixed line-feed printer- Using file #'s 128-255
Printing garbage	<ul style="list-style-type: none">- Sending graphics in mode 4 or 5- Switch 3 off- Wrong printer selected
Interface not paying attention to command in OPEN	<ul style="list-style-type: none">- Interface locked, try 'U' in command channel
Text and graphics different widths	<ul style="list-style-type: none">- Send 'IY' to command channel

APPENDIX B Special codes

All the following special commands are exactly the same as the ones for the VIC-1525 printer.

CHR\$(8)

This command puts the interface into the graphics mode. If sent while already in the graphics mode, code 8 will be passed to the printer.

CHR\$(14)

This is the command to start printing double-width. If in the graphics mode, this will terminate it. If you want to send literally code 14 to your printer, send instead the code for your printer listed in table B1. The interface will convert it to 14.

CHR\$(15)

This is the command to stop printing double-width. If in the graphics mode, this will shut it off. If you want a 15 to get to your printer, send the code for your printer in table B1 and it will be changed to 15 for you.

CHR\$(16)

This command is used to specify where to start printing next. For example, to print 'HELLO' starting at the 19th column, send CHR\$(16);"19HELLO".

CHR\$(27);CHR\$(16);CHR\$(n1);CHR\$(n2)

This command, only effective while in the

CHR\$(8) graphics mode, specifies which dot column to start printing at next (number of dots instead of number of characters). The number of dots is calculated by $n1 + n2 \times 256$. For example, to print an asterisk 300 dots from the left margin, use CHR\$(27);CHR\$(16);CHR\$(44);CHR\$(1);"*" (44 + 1*256=300).

CHR\$(26);CHR\$(n);CHR\$(g)
This command also can only be used while in the graphic mode. It repeats the graphics dot column 'g', n times. Remember that bytes specifying dot columns in the graphics mode must have their top bit set (must be ≥ 128). (eg. CHR\$(8)CHR\$(26)CHR\$(200)CHR\$(255) will print a bar 200 dots long.)

CHR\$(17)
Cursor down mode (upper/lower case).

CHR\$(145)
Cursor up mode (upper case only).

CHR\$(18)
Turn on reverse field mode (white on black printing).

CHR\$(146)
Turn off reverse field mode.

SWITCHES			TO GET	TO GET
A	B	C	14 SEND	15 SEND
off	off	off	14	20
on	off	off	31	30
off	on	off	14	15
on	on	off	14	15

Table B-1

APPENDIX C
Listing abbreviations

ASCII	SYMBOL	KEY
5	WHT	White
8	DISH	Disable shift
9	ENSH	Enable shift
14	SWLC	Switch to lower
17	DOWN	Cursor down
18	RVON	RVS on
19	HOME	Home
20	DEL	Delete
28	RED	Red
29	RGHT	Cursor right
30	GRN	Green
31	BLU	Blue
129	ORNG	Orange
133	F1	Function F1
134	F3	Function F3
135	F5	Function F5
136	F7	Function F7
137	F2	Function F2
138	F4	Function F4
139	F6	Function F6
140	F8	Function F8
141	SHRT	Shift Return
142	SWUC	Switch to upper only
144	BLK	Black
145	UP	Cursor up
146	RVOF	RVS off
147	CLR	Clear
148	INST	Insert
149	BRWN	Brown
150	LRED	Light red
151	GRY1	Gray 1
152	GRY2	Gray 2
153	LGRN	Light Green

154	LBLU	Light blue
155	GRY3	Gray 3
156	PUR	Purple
157	LEFT	Cursor left
158	YEL	Yellow
159	CYN	Cyan

APPENDIX D Parallel cable pinout

PIN#	FUNCTION
1	Data strobe (active low)
2	Data bit 0
3	Data bit 1
4	Data bit 2
5	Data bit 3
6	Data bit 4
7	Data bit 5
8	Data bit 6
9	Data bit 7
10	Acknowledge (active low)
31	Printer reset (active low)
16, 19-30, 33	Grounded
All others	No connection

APPENDIX E Interfacing non-standard printers

If you cannot find your printer listed in the table on page 3 then your interface will not automatically support it. There is a way, however, to 'customize' the interface to make it compatible with your printer. The procedure involves sending a string of 18 bytes through the command channel which gives your interface vital information about your

printer. Call Xetec for information on the codes necessary for your particular printer; more are being compiled all the time.

APPENDIX F Changing the device number

The state of switch number 1 at power-up (or reset) determines the interface's device number (off = 4, on = 5). From software, you can change the device number to anything from 0 to 31. Send this to the command channel:

```
PRINT#<file>,"D"CHR$(device)
```

NOTE: setting the device number in this way is only temporary; it will change back to 4 or 5 when the interface is reset or powered-up again. Also, keep in mind that some numbers may cause a conflict with other devices (such as 8 with the disk) depending on your system.

APPENDIX G Sample programs

```
10 REM WORD PROCESSOR LOCK PROGRAM
20 REM LOCK IN UPPER/LOWER CASE WITH L.F.
30 OPEN 1,4,27:REM READY TO LOCK IN MODE 7
40 PRINT#1:REM YOU MUST PRINT SOMETHING TO
50 REM COMPLETE THE LOCK.
60 CLOSE 1:END
```

```

5  LISTING DEMO
10 OPEN 4,4
20 OPEN 1,4,15:REM COMMAND CHANNEL
30 PRINT#1,"MG":PRINT#4,"DEFAULT MODES:"
40 CMD4:LIST 140-
50 PRINT#1,"V":PRINT#4,"V MODE"
60 CMD4:LIST 140-
70 PRINT#1,"A":PRINT#4,"A MODE"
80 CMD4:LIST 140-
90 PRINT#1,"K":PRINT#4,"K MODE"
100 CMD4:LIST 140-
110 PRINT#1,"MG":PRINT#4
120 CLOSE 1:CLOSE 4
130 END
140 PRINT"[HOME][DOWN][DOWN][DOWN]"
150 PRINT"      "

```

```

10 REM VIC-20 SCREEN DUMP
20 REM TO PRINT AN IMAGE OF THE SCREEN,
30 REM USE GOSUB 63000 (OR 63001 FOR
40 REM AN INVERSE DUMP). INCLUDE THESE
50 REM LINES IN YOUR PROGRAM:
63000 RV=0:GOTO 63010
63001 RV=1
63010 A=PEEK(36869)
63020 SM=PEEK(36866)AND128)*4+(A AND 112)*64
63030 IFA>239THENGGM=32768:GOTO63080
63040 A=A AND 15
63050 IF A<4 THEN GM=32768+A*1024:GOTO63080
63060 IF A>11 THEN GM=A*1024-8192:GOTO63080
63070 PRINT"ERROR":STOP
63080 CLOSE99:OPEN 99,4
63090 PRINT#99:PRINT#99
63100 FORX=175TO0STEP-7:FORY=0TO22
63110 XS=X AND 7
63120 C=SM+Y*22+INT(X/8)

```

```

63130 C1=PEEK(C):C2=32:IFX>7THENC2=PEEK(C-1)
63140 FORI=0TO7
63145 GW=PEEK(GM+8*C2+I)*256+PEEK(GM+8*C1+I)
63150 GB=INT(GW/(2 (7-XS)))
63153 GB=GB-128*INT(GB/128)
63155 IF RV=1 THEN GB=255-GB
63160 PRINT#99,CHR$(GB OR 128);:NEXTI,Y
63170 PRINT#99:NEXTX
63180 PRINT#99:CLOSE4:RETURN

```

```

10 REM C-64 HIRES SCREEN DUMP
20 REM RUN THIS PROGRAM TO LOAD THE MACHINE
30 REM LANGUAGE DUMP ROUTINE. POKE THE START
40 REM ADDRESS OF THE BIT MAP/256 IN LOCATION
50 REM 53244 THEN SYS 52992
60 AD=52992
70 READ A$:FOR J=1 TO 35 STEP 3
80 V$=MID$(A$,J,1):GOSUB 1000:P=V*16
90 V$=MID$(A$,J+1,1):GOSUB 1000:P=P+V
100 POKE AD,P:AD=AD+1
110 NEXT J:GOTO 70
130 DATA A9 00 20 BD FF A9 04 A2 04 A0 FF 20
140 DATA BA FF 20 C0 FF A2 04 20 C9 FF EA EA
150 DATA EA EA EA EA EA EA A9 08 20 D2 FF EA
160 DATA EA A0 00 A2 00 8E F9 CF A9 01 8D FA
170 DATA CF A9 06 20 76 CF 38 E9 01 10 F8 E8
180 DATA AD FA CF 20 D2 FF AD F9 CF F0 06 E0
190 DATA 40 B0 0D 90 DF E0 00 D0 DB A9 01 8D
200 DATA F9 CF D0 D4 A9 0D 20 D2 FF 98 18 69
210 DATA 07 A8 C0 C7 90 C1 A2 03 20 C9 FF 60
220 DATA EA EA EA EA EA EA EA EA EA EA 8D FB
230 DATA CF 98 48 8A 48 98 18 6D FB CF A8 A9
240 DATA 00 85 FC C0 C8 90 06 18 90 5F EA EA
250 DATA EA 98 29 F8 85 FB 06 FB 26 FC 06 FB
260 DATA 26 FC 18 65 FB 85 FB A9 00 65 FC 85
270 DATA FC 06 FB 26 FC 06 FB 26 FC 06 FB 26
280 DATA FC 8A 29 F8 18 65 FB 85 FB AD F9 CF

```

```
290 DATA 65 FC 85 FC 98 29 07 18 65 FB 85 FB
300 DATA AD FC CF 65 FC 85 FC 8A 29 07 AA E8
310 DATA A9 00 38 6A CA D0 FC 8D FD CF A0 00
320 DATA B1 FB 18 2D FD CF F0 01 38 2E FA CF
330 DATA 68 AA 68 A8 AD FB CF 60 EA 01 80 00
340 DATA 20 01 EA 77 *
1000 IF V$="*" THEN STOP
1010 V=ASC(V$)-48:IF V>10 THEN V=V-7
1020 RETURN
```

A PERSONAL WORD

Thank you for purchasing one of our products. We at Xetec have made every effort to deliver to you a quality product that will be a real asset to your computing. You are a real asset to us, too. Please do not hesitate to contact us with your questions or suggestions, either at the address below or by calling us at (913) 827-0685 during normal business hours (central time).

NOTICE OF LIMITED WARRANTY

Xetec, Inc. warrants this product against defects in material and/or workmanship for a period of one year from the date of purchase and will, during this warranty period, repair or replace any defective components, or, at our option, the entire product, at no charge, provided that the product is returned shipping prepaid with proof of original purchase to:

Xetec, Inc.
3010 Arnold Rd.
Salina, KS. 67401

This warranty does not apply if, in Xetec's opinion, the product has been damaged by accident, misapplication, misuse, or as a result of service, modification, or adjustment, by other than the manufacturer, Xetec, Inc. No other warranties are expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Xetec, Inc. is not responsible for consequential damages of any kind arising from or connected with the use of this product.

VIC-20 and C-64 are trademarks of Commodore Business Machines, Inc.

Copyright (C) 1984 Xetec, Inc.
Printed in USA